

Research Article

Posture Optimization Algorithm for Large Structure Assemblies Based on Skin Model

Wei Zhang ^{1,2}, Luling An ¹, Peter Sherar,² and Wei Tian¹

¹College of Mechanical and Electrical Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, 210016, China

²School of Aerospace, Transport and Manufacturing, Cranfield University, MK43 0AL, Bedford, UK

Correspondence should be addressed to Luling An; anllme@nuaa.edu.cn

Received 15 June 2018; Revised 31 August 2018; Accepted 20 September 2018; Published 18 October 2018

Academic Editor: Oliver Schütze

Copyright © 2018 Wei Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Geometric deviations inevitably occur in product manufacturing and seriously affect the assembly quality and product functionality. Assembly simulations on the basis of computer-aided design (CAD) package could imitate the assembly process and thus find out the design deficiencies and detect the assemblability of the components. Although lots of researches have been done on the prediction of assembly variation considering the geometric errors, most of them only simplify the geometric variation as orientation and position deviation rather than the manufacturing deformation. However, in machinery manufacturing, even if the manufacturing defects are limited, they could propagate and accumulate through components and lead to a noncompliant assembly. Recently, many point-based models have been applied to assembly simulation; however they are mainly interested in simulating the resulting positions of the assembled parts and lack the consideration of the postprocessing after positioning. This paper enriches the complete assembly simulation process based on skin model and presents a simple and effective posture evaluation and optimization method. The studied approach includes a software algorithm applied to evaluate the contact state of the assembly parts and a mathematical model based on the particle swarm optimization to acquire the optimal assembly posture. To verify the efficiency and feasibility of the proposed method, a case study on the aircraft wing box scaling model assembly is performed.

1. Introduction

Traditional assembly simulation aims at evaluating the assemblability of the components and to predict the interference between assembly parts, which is normally based on the nominal CAD models. The virtual assemblies so obtained have the limitation of considering the nominal geometry only. In order to calculate the assembly variation more accurately, many researchers have devoted their efforts to modeling and analyzing the assembly variation on the basis of geometric deviations. Monte Carlo simulations are commonly used in tolerance analysis [1, 2], but this method needs massive samples to obtain reliable results [3]. T. Zhang et al. [4] used the principal component analysis (PCA) to extract the common features of a batch of parts and provide a statistic model of the geometric form errors, which could be applied as a guide in assembly process optimization. E. Goka et al. [5] decomposed the shape model into a nominal surface plus modal vectors and calculated the assemblability

of two components through mathematical optimization. The real features in these models are however simplified and represented by parameters, such as plane and cylindrical surface features. W. Polini et al. [6] carried out the tolerance analysis with Jacobian model and obtained similar results to the vector loop model. However, this model is only suitable when translational variations are involved. All the above geometric variation models only consider the orientation and position deviations of the nominal surface rather than the form errors.

The skin model is a basic concept within Geometrical Product Specification (GPS) and verification and attracts more and more attention in computer-aided tolerance (CAT) analysis [7, 8]. Point clouds and surface meshes are the virtual representations of skin model shape. Figure 1 shows the difference between the theoretical model and the skin model which is obtained from the manufactured part. M. Zhang et al. [9] and N. Anwer et al. [10] first focused on the skin model representation and simulation. L. Homri et

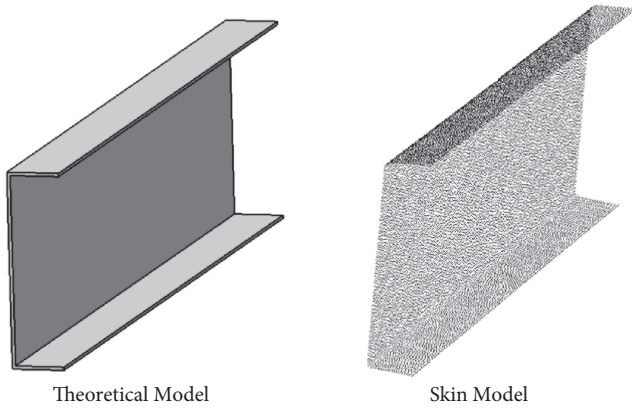


FIGURE 1: The difference between theoretical model and skin model.

al. [11] proposed a new method to generate form defects of cylinders and converted the assembly simulation problem into a relative positioning problem between the difference surface and one of the perfect surfaces. But it mainly focused on modeling form defects by metric modal decomposition and the tolerance analysis of cylindrical parts rather than the whole assembly simulation process. Then, B. Schleich et al. [12] pointed out the application of the skin model in mechanical engineering and the challenges in assembly simulation techniques. They also proposed two different approaches for the relative positioning of point cloud models to simulate the assembly process of skin model shapes [13], which makes a first step towards a comprehensive CAT approach employing point-based models. After that, B. Schleich et al. [14] extended the skin model into the contact assembly simulation and analyzed the contact quality and kinematic behavior. Later, a general framework for the assembly simulation approaches which used an iterative procedure to execute different registration methods was proposed [15]. This general framework is mainly based on [13] but allows the combination of different registration approaches. Although the aforementioned researches enriched the mechanical assembly with form errors, the multiple-part assembly situation is not covered in the simulation model and the postprocessing after point cloud positioning is not mentioned as well.

Posture evaluation is a necessary task in large structure assembly, which is aimed at calculating the posture deviations of the assembled part and using the calculated results to guide the position and orientation adjustment of the components. B. Zhang et al. [16] proposed a trajectory planning method based on a novel six-degree-of-freedom posture alignment system, and the displacement and actuating torque of each joint of the positioners were simulated as well. Y. LI et al. [17] obtained the rotation and translation parameters of posture adjustment by the proposed multiobjective optimization algorithm based on the Gauss-Newton method. H. Yukan et al. [18] introduced a relatively inexpensive method for posture evaluation of the positioning of the wing-body assembly, and the results showed that positioning was an essential process to guarantee alignment accuracy in an assembly line. W. Tang et al. [19] presented the wing posture evaluation based on

the iterative closest point algorithm, and the wing posture evaluation problem was transformed into a registration problem between space points and surface. Based on the existing literatures, posture evaluation methods and the approaches of reducing posture calculation errors are addressed by many scholars, but the geometric deviations of the assembled part are not taken into consideration.

This paper enriched the complete assembly simulation process based on skin model. Not only was the numerical solution of the posture optimization problem presented in detail, but also the contact states of the assembled parts as well as the assembly gaps are evaluated quantitatively, which could be applied as a guide in the assembly postprocessing (shimming or fettling) to improve the assembly efficiency and accuracy. The profiles of mating surfaces are measured and represented by point clouds. The assembly deviation distribution between multiple components is investigated quantitatively. The structure of this paper is organized as follows. Section 2 illustrates the assembly simulation of multiple components based on the skin model. The general approaches for the positioning of skin model are discussed and an improved point cloud registration technique is proposed. After that, a particle swarm optimization (PSO) algorithm is developed to realize the posture adjustment. Section 3 presents the assembly interference calculating process between mating surfaces in order to facilitate later assembly process, such as shimming or fettling. A case study is conducted in Section 4 to demonstrate the implementation procedures, including the preprocess of the point clouds, mating surfaces extraction, posture adjustment, and final shape evaluation. Finally, a brief summary is given in Section 5.

2. Assembly Simulation of Multiple Components Based on Skin Model

In this section, the skin models of the manufactured parts are firstly positioned according to their assembly process. Then a constrained optimization algorithm on the basis of particle swarm optimization (PSO) is presented to adjust the part posture and the transformation matrices of adjustment are obtained. In order to describe the algorithm conveniently, some terms are defined as follows:

- (1) Assembly coordinate system (ACS) is termed as $O\text{-}XYZ$ and it is the global system of assembly process
- (2) Measurement coordinate system (MCS) is the default coordinate system of optical facilities and can be defined as $O_M\text{-}X_M Y_M Z_M$
- (3) Local coordinate system (LCS) for the movable components during assembly is known as $O_L\text{-}X_L Y_L Z_L$. It is defined in the digital model of movable components

Classification of part postures:

- (1) Theoretical posture is the position and orientation of the digital model in the ACS
- (2) Measurement posture is acquired through the three-dimensional laser scanner in the MCS

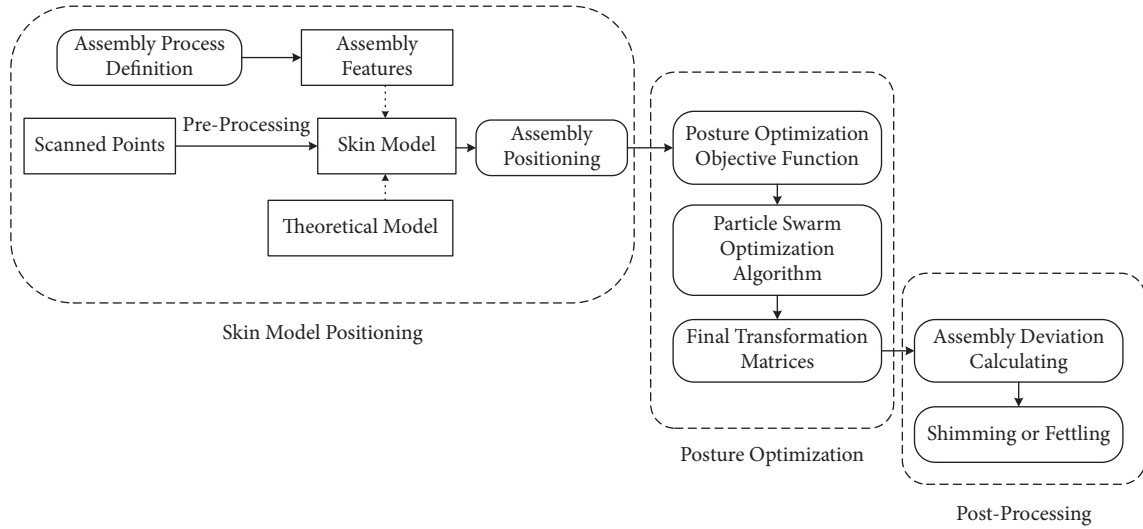


FIGURE 2: The assembly simulation and posture optimization process.

- (3) Target posture, i.e., the final state of the panel, is set according to the adjusting result

The complete procedure of assembly simulation and posture optimization for mechanical assemblies consists of three main steps, i.e., the skin model positioning of the assembled parts, the software algorithm of the posture optimization, and the assembly postprocessing, as demonstrated in Figure 2. Based on the PSO algorithm of the posture optimization, the final adjustment parameters can be computed and output by processing a set of input data, and the posture of the actual part could be adjusted by regulating the part locators according to the parameters.

2.1. Positioning Method of Skin Model. The assembly simulation conducted in various CAD packages such as CATIA, UG, and SolidWorks is usually worked as a detection tool of product design, which is normally based on a nominal CAD model measured in ACS. They are not capable of dealing with discrete geometry model and do not allow the consideration of form errors. The skin model could however reflect the real surface of the manufactured part but the application in assembly simulation is still a challenge.

The positioning of skin models is based on the registration of their point clouds. The iterative closest point (ICP) method introduced by P. Besl et al. [20] is widely used for the registration of 3D point clouds to theoretical models. It repeats the process of searching for the nearest point and calculating its optimal parameters until the convergence criteria are well satisfied. However, it is time-consuming to find the nearest point when the model contains millions of points and the algorithm may easily get trapped in a local optimum. Therefore, an improved point clouds registration method was proposed in this paper. Since the skin model is measured in MCS which is always far away from the assembly position of nominal part in ACS, a coordinate transformation is needed to unify coordinate systems. After that, we use the assembly features on the point cloud as key characteristics to

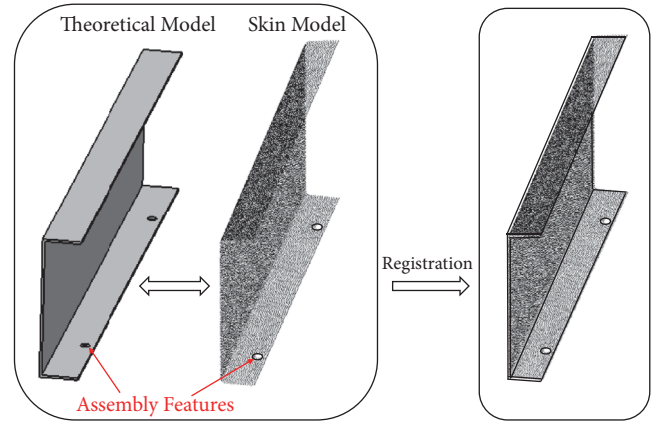


FIGURE 3: Skin model positioning based on assembly features.

define the initial position of the skin model. The assembly features are an associated set of geometric factors that have the capability of generating assembly constraint (such as aligned, fitted, coaxial and tangent, etc.) [21]. The geometric characteristics that form the assembly features mainly involve points, lines, and planes. For instance, in Figure 3 the two assembly holes on the flange are used for the positioning of the part. Thus, according to the assembly process, these two cylindrical holes on the skin model could be fitted and aligned with the nominal model to obtain the theoretical posture. Based on the above, the ICP algorithm could be applied to finish the registration of the skin model. For the detailed ICP algorithm principle, the reader can refer to these literatures: P. Besl et al. [20]; S. Kaneko et al. [22].

2.2. Posture Optimization Algorithm Based on the Skin Model. Due to the form errors on the component surfaces, the interference or gaps inevitably occur between the mating surfaces. Although special clearance gauges are used to

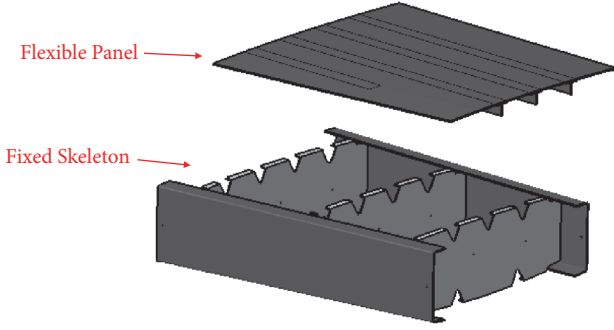


FIGURE 4: Wing box panel assembly.

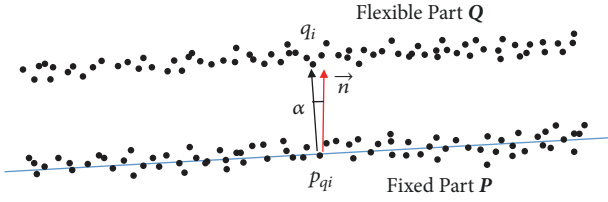


FIGURE 5: The point clouds of two assembled parts.

quantitatively measure the gaps, the handling process is relatively of low efficiency. Not only are highly skilled engineers required, but also multiple trials are needed to achieve the desired shape. To eliminate the gaps between the mating surfaces, forces are applied to deform the part, which could induce assembly stresses around the structure and impair the reliability. Alternatively, shims could be used when the gaps are relatively large [23]. Therefore, if the assembly deviations between components could be predicted quantitatively, then the posture of the assembly part could be optimized according to the calculated deviations so as to reduce the overall gaps or even eliminate the gaps.

(1) *Mathematical Modeling for the Assembly Process.* The main idea behind the posture optimization of multiple assembly parts is based on the computation of point clouds. Here the assembly process is assumed as follows: one adjustable part is assembled onto multiple fixed parts, such as the panel assembled onto the skeleton illustrated in Figure 4. The data point clouds of those parts are obtained by optical measurement systems and defined as flexible part $Q = \{q_i\}_{i=1}^M$ and fixed parts $P = \{P_1, P_2, \dots, P_H\}$, $P_1 = \{p_i\}_{i=1}^N$, where M and N are the number of points on the part and H is the number of fixed parts. Then the assembly process is that parts P stay fixed and flexible part Q is moved and adjusted to fit the parts P . Part Q is located on the measurement posture before adjusting, and the target posture of Q could be obtained by minimizing an objective function E , which is a function of the rigid body transformation of flexible part Q and the fixed parts P . That is,

$$\min E(\mathbf{k}(Q), P). \quad (1)$$

Since part Q is seen as rigid during the adjustment, only translation matrix T and rotation matrix R are involved in

this problem. Therefore, the rigid body transformation of part Q could be written as

$$\mathbf{k}(Q) = Q + (T + R \times Q), \quad T, R \in \mathbb{R}^3. \quad (2)$$

Many problems in geometry processing are stated as least-squares optimizations. Although least-squares problems are widely used, they have immanent drawbacks such as high sensitivity to outliers. Flöry and Hofer [24] applied the l_1 -norm $\|r\|_1 = \sum_i |r_i|$ to solve the optimization problem. Based on the discussions and comparisons with least-squares methods which are commonly used on point clouds fitting and registration, it points out that minimizing the unsigned distance function yields more robust optimization problems. Moreover, the signed distance can be seen as an indicator of whether or not the assembled parts collide. Therefore, a modified method of minimizing the signed distance function was applied in the research work.

In order to consider the assembly process and the actual assembly situation, constraints need to be added onto the transformation matrices T and R . The optimization purpose is decreasing the overall gaps between the mating surfaces and the assembly process is adjusting the position of Q to fit P . Here we define the clearance distance between two point clouds as d . For a point cloud Q which contains M points, the overall clearance could be represented as the sum of the distance and thus the objective function can be written as

$$\min_{T,R} E(\bullet) = \sum_{i=1}^M d(\mathbf{k}(q_i), p_{qi}, \mathbf{w}), \quad \forall q_i \in Q, \quad (3)$$

where q_i is the point on part Q , p_{qi} means the corresponding nearest point on parts P , and $\mathbf{w} = [w_1 \ w_2 \ w_3]^T$ is the normal vector of the mating surface on fixed parts P , as illustrated in Figure 5. Here we use the nearest neighbor to establish the relationship between the two point clouds; that is, for each point on the flexible part q_i , the corresponding point on the fixed parts p_{qi} is determined on the basis of Euclidean distance d_E , for example, $d_E(\mathbf{k}(q_i), p_{qi}) = \|\mathbf{k}(q_i) - p_{qi}\|$. \mathbf{w} is worked as a weighting factor to determine the value of interference or gaps and it could also be used as a contact quality evaluation. Therefore, in this objective function, the value of clearance between two point clouds is represented by weighted distance between the adjusted point $\mathbf{k}(q_i)$ and p_{qi} , that is, $d(\mathbf{k}(q_i), p_{qi}, \mathbf{w}) = |(\mathbf{k}(q_i) - p_{qi}) \cdot \mathbf{w}|$.

In order to check the contact status of the mating surfaces, the signed weighted distance d_s is calculated and worked as a constraint to avoid interference, that is, $d_s(\mathbf{k}(q_i), p_{qi}, \mathbf{w}) = (\mathbf{k}(q_i) - p_{qi}) \cdot \mathbf{w}$. Whereas the negative value indicates the interpenetration between assembled parts, the positive value means the gaps between them. In this regard, the final optimization objective function becomes minimizing the overall clearance distance between the multiple parts under certain constraints. That is,

$$\min_{T,R} E(\bullet) = \sum_{i=1}^M d(\mathbf{k}(q_i), p_{qi}, \mathbf{w}) \quad (4)$$

$$\text{subject to } d_s(\mathbf{k}(q_i), p_{qi}, \mathbf{w}) \geq 0, \quad \forall q_i \in Q. \quad (5)$$

(2) *Particle Swarm Optimization Algorithm*. There are many metaheuristic algorithms that can be applied to solve various optimization problems, such as evolutionary algorithms, ant colony optimizations, artificial immune systems, and neural networks [25]. And these schemes gain lots of advantages compared to conventional linear and nonlinear optimization techniques. Particle swarm optimization (PSO) is a population-based stochastic algorithm, which was proposed by R. Eberhart et al. [26] and modified by Y. Shi et al. [27]. The main idea is prompting the movement of the whole group to evolve from disorder to order through sharing the information of all population members, so as to find the optimal solution of the problem [28]. PSO is an evolutionary algorithm based on iteration and it does not need the gradient information since it is based on probability searching [29]. Compared with other swarm intelligence (SI) optimization methods such as ant colony optimization (ACO) [30], shuffled frog leaping algorithm (SFLA) [31], and artificial bee colony (ABC) algorithm [32], the underlying idea behind PSO is similar to other SI algorithms [33]. However, PSO has easier implementability and lower computational complexity. The more vital reason is that the solution of PSO depends minimally on the initial states of particles. Thus, PSO is a good option to solve the posture optimization problem. Besides, the effectiveness of PSO in engineering optimization problems has been verified by many researchers [15, 29, 34, 35].

In this problem, we define the unsolved posture as a particle and the six parameters ($T_x, T_y, T_z, \alpha, \beta, \gamma$) are the dimensions of the particle. Assuming $\mathbf{X}_i = (x_{i1}, x_{i2}, \dots, x_{i6})$ is the position vector of particle i and $\mathbf{V}_i = (v_{i1}, v_{i2}, \dots, v_{i6})$ is the velocity vector of particle i , then, the modified iterative operator of PSO can be written as follows:

Velocity vector iterative formula:

$$\mathbf{V}_i = w\mathbf{V}_i + c_1 r_1 (\mathbf{P}_{best_i} - \mathbf{X}_i) + c_2 r_2 (\mathbf{G}_{best} - \mathbf{X}_i), \quad (6)$$

position vector iterative formula:

$$\mathbf{X}_i = \mathbf{X}_i + \mathbf{V}_i, \quad (7)$$

where \mathbf{P}_{best_i} and \mathbf{G}_{best} represent the history optimal position of particle i and the global history optimal position of the population, respectively. Parameter w is the inertia weight which has the value within $[0, 1]$. Generally, w is defined through the adaptive value method. At the beginning, w is set relatively large to gain a better global optimization ability; then w decreases gradually to enhance the local optimization ability. Here the inertia weight w is empirically defined as

$$w = 0.9 + (0.9 - 0.5) \times e^{((-20 \times i^6)/maxgen^6)}, \quad (8)$$

where i is the number of generation and $maxgen$ is the maximum value of generation.

c_1 and c_2 are positive acceleration constants that represent the learning ability, which can be empirically defined as

$$c_1 = 2.5 - 1.5 \times \frac{i}{maxgen} \quad (9)$$

$$c_2 = 1.0 + 1.5 \times \frac{i}{maxgen}. \quad (10)$$

r_1 and r_2 are independent random values within $[0, 1]$. The aforementioned key parameters were empirically derived from continual testing and verification. However, the theoretical basis of parameter setting needs more research in the future work.

Another key issue that needs to be solved in this optimization problem is that the objective function has an inequality constraint which was imposed to avoid interference between assembled parts. Since PSO is primarily applied on unconstrained optimization problems, a constraint-handling technique was added. There are several approaches that can handle constraints within the framework of PSO algorithm. For example, many penalty-based PSO approaches have been developed and examined in literatures [36–38]. Sedlaczek and Eberhard [39] proposed an augmented Lagrangian PSO algorithm which combined the conventional PSO with the augmented Lagrangian multiplier. According to the optimization problem presented in our research work, a multi-level penalty function [40, 41] based method was adopted to transform the constraint problem into an unconstrained problem. A modified objective function F is applied, which turns the inequality constraint problem as follows:

$$F(x) = f(x) + \sum_{i=1}^n q_i \times S \times (g_i(x))^2, \quad (11)$$

where $f(x)$ is the objective function, S is a penalty imposed for violation of the constraint, which is chosen based on preliminary trials of the algorithm and here is set to be 1000, n is the number of constraints, and $g_i(x)$ is the inequality constraint. If the constraint is violated the value of q_i will be one; otherwise, it will be zero.

The procedure of the PSO algorithm is the particle searching their position and velocity according to the fitness of the corresponding objective function and the constraints to find the optimal solution. The calculating procedure of the PSO algorithm is as follows.

Step 1 (population initialization). After random initialization, calculate the fitness so as to find the optimal position \mathbf{P}_{best_i} of particle i and the global optimal position \mathbf{G}_{best} .

Step 2 (iteration setting). Set evolution times g_{max} and initialize the current iteration $g = 1$.

Step 3 (speed update). Update the velocity vector of every particle according to (6).

Step 4 (position update). Update the position vector of every particle according to (7).

Step 5 (local and global position update). Update \mathbf{P}_{best_i} of particle i and the global position \mathbf{G}_{best} .

Step 6 (judgment of termination conditions). If the solution satisfies accuracy requirement or the iteration time satisfies g_{max} then output \mathbf{G}_{best} . Otherwise, go on calculating and go back to Step 3.

After every iteration, the points on part Q will update their positions until satisfying the termination conditions. Then, the final optimal position of part Q will be stored and the rigid transformation matrices from the measurement posture to target posture will be calculated. Assuming the rotation angles on the X , Y , and Z axes as α , β , and γ , respectively, then the rotation matrices can be written as

$$\begin{aligned} R_x &= \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}, \\ R_y &= \begin{bmatrix} \cos \beta & 0 & -\sin \beta \\ 0 & 1 & 0 \\ \sin \beta & 0 & \cos \beta \end{bmatrix}, \\ R_z &= \begin{bmatrix} \cos \gamma & \sin \gamma & 0 \\ -\sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}. \end{aligned} \quad (12)$$

The final rotation matrix can be derived from matrix concatenation: $R=R_x R_y R_z$, and the translation matrix can be expressed as $T=[T_x \ T_y \ T_z]$. Therefore, the final adjustment parameters can be calculated through processing a set of input data; then the posture of the actual part could be adjusted by regulating the part locators according to the parameters.

3. Assembly Deviation Calculation

Generally, the gaps between mating surfaces could be assessed using special gauges or other manual inspection techniques; then, based on the detection results, components need to be shimmed or fettled to maintain the assembly tolerances. However, this detection process is relatively time-consuming and of low efficiency. Based on the aforementioned iteration algorithm in Section 2, the contact state of the two surfaces could also be evaluated and the gaps could be computed quantitatively. Thus, the distribution of interference could be predicted and the bespoke interfaces mentioned by P. Maropoulos et al. [42] could be formed according to the calculating results, so that those parts could fit to one another before physically assembling them. The main point in this section is proposing a simple and effective method to calculate the assembly interference in order to provide a basis for forming the bespoke interfaces.

After the posture optimization process, all the assembled parts are positioned on the optimal position. Then, the assembly deviations between the multiple parts could be evaluated. Parts can only generate assembly deviations between mating surfaces, which means other surfaces are not involved in the assembly and do not affect the deviation results. Hence, in order to simplify the calculation, only the mating surfaces are extracted. Figure 6 shows the extraction of the skin model for the assembly interference calculation. In order to partition the interface from the point cloud,

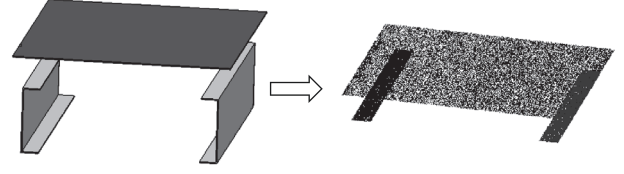


FIGURE 6: Mating surfaces extraction.

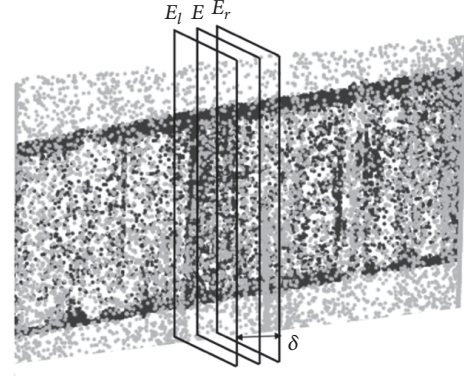


FIGURE 7: Point cloud slicing diagram.

which is dense, nonhomogenous, and noisy [43], a boundary extraction method based on normal vector is applied. The points whose normal vectors change sharply will be extracted to form the boundary points. The determination of the surface normal is approximated to the estimation of the tangent surface normal, which could convert to a least-squares fitting problem. Thus, the final calculation of the surface normal could be solved by analyzing the eigenvector and eigenvalue of the covariance matrix, that is, the principal component analysis method introduced by W. Dunn et al. [44].

After extracting the mating surfaces of the assembled parts, an interference calculation method based on the point cloud slicing is proposed. As skin models are dense and discrete points, it is hard to get a complete section curve on one slice plane. Thus, a section step size δ is set to calculate the section curve. Following the direction of the normal vector on slice plane E , parallel planes E_l and E_r are generated from an equidistance of $\delta/2$, as shown in Figure 7. Then points located in each data region (between E_l and E_r) are chosen as valid points to generate the section curve by projecting them onto plane E . The detailed calculating procedures are illustrated in Figure 8. Based on the step size δ , points in each data region could be divided into two neighbor sets: K_l and K_r . For any point P_{li} in K_l , a corresponding nearest neighbor P_{ri} in K_r could be found according to the Euclidean distance and these two points could form a line l . Then the intersection point q_i (x_i, y_i, z_i) of l and E is saved to form the section curve. For each slice plane, the deviation between two section curves could be calculated; therefore, the bespoke interfaces could be formed.

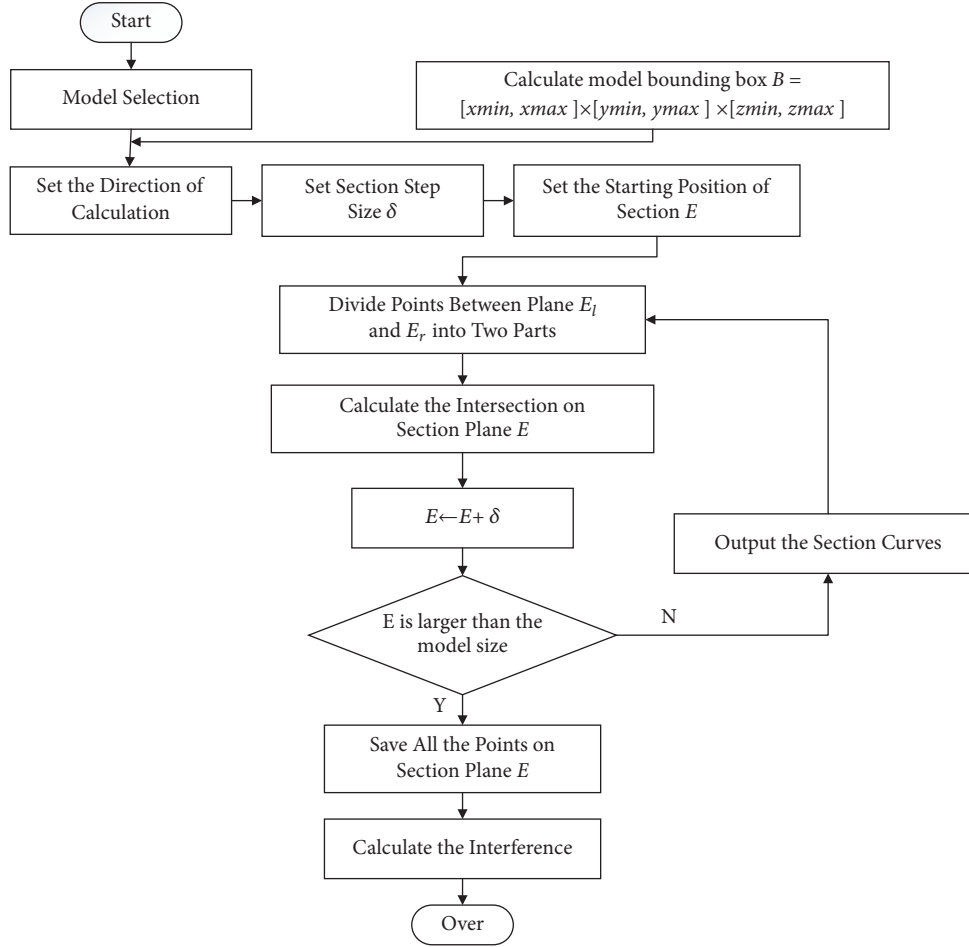


FIGURE 8: Assembly deviation calculation procedures.

4. Case Study

The proposed posture optimization algorithm and assembly deviation calculating method has been applied to a real aircraft wing box scaling model assembly process to validate the feasibility and efficiency. The structure of the wing box consists of two main parts, i.e., the skeleton, which contains ribs and spars, and the panel. The posture optimization method proposed in this paper is applied on the adjustment of the panel. The main steps of the assembly simulation and posture optimization experiment consist of the positioning of the assembled parts, the posture optimization, and the assembly postprocessing, which have been discussed in Section 2 in detail.

4.1. The Preprocess of the Skin Model. The skin models of the manufactured components were obtained by the FARO Edge 2.7 optical measurement system in this experiment. Since scanned points are dense and noisy, some preprocesses of the point clouds need to be conducted.

- (1) Distorted point removal: Since the data point clouds were obtained through optical devices, the background points are inevitably scanned and cause some

noise. These points must be filtered out before the analysis

- (2) Data reduction: The number of scanned points could be up to millions and huge computing cost will be needed to handle the mass points. However, the more points do not mean the higher computing accuracy, so appropriately simplified point clouds could maintain the calculating precision. Furthermore, data reduction improves the computing efficiency and hence saves cost
- (3) Multiview point cloud registration: Since the manufactured parts are large and complex, we could not obtain the whole surfaces of the model by one scan. Thus, the multiview registration is implemented to get a complete model, which could be realized by the common characteristic points registration on different point cloud pieces

After the preprocess of the point clouds, the skin models of the manufactured parts could be obtained. Then the multiple parts are located according to the assembly process and the coordinate systems are transformed from MCS to ACS.

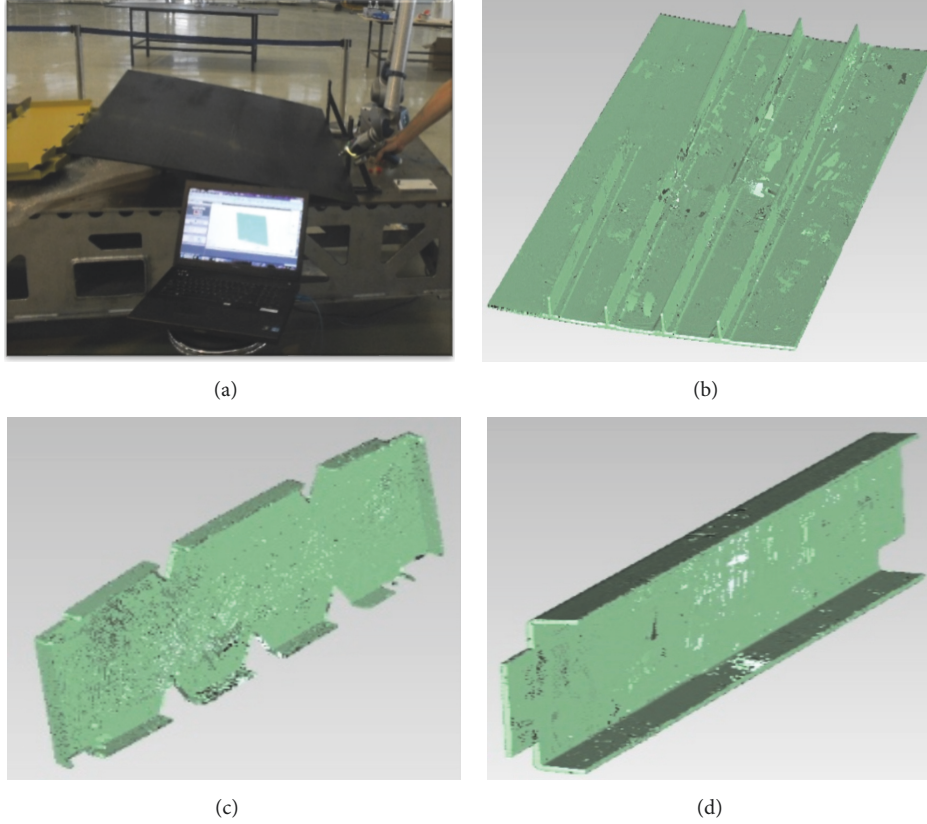


FIGURE 9: The skin model acquisition: (a) scanning process, (b) the panel, (c) the rib, and (d) the spar.

Figure 9 shows the scanning process and the corresponding skin models of the panel and skeleton.

4.2. PSO Algorithm Improvement. During the wing box assembly process, the skeleton is set as assembly datum and the panel is moved and adjusted to fit the skeleton. The signed weighted distance d_s between the skeleton and the panel is calculated as the objective function to reduce the overall gaps between them. However, since the particle in PSO has no selection mechanism, when the particle swarm is concentrated around a local extremum, prematurity often occurs. In order to avoid the prematurity, a hybrid GA-PSO algorithm was proposed where the genetic operators, i.e., crossover and mutation, are incorporated into the PSO algorithm. Then, the effectiveness of the algorithms before and that after improvement were compared in the experiment.

According to Garg [41], GA is good at reaching a global region, but the weakness is that if an individual is not selected then the information contained by that individual is lost. PSO is good at searching for an optimal solution with the help of group interactions, but without a selection operator, PSO may waste resources on poor individuals. In this experiment, the genetic operators are embedded into the PSO algorithm. The mutation operator can enable the population to generate new individuals through adjustment, and the overall diversity can be guaranteed. The crossover operator can get new individuals by crossing and then enlarges the search range,

which plays an important role in the global optimal solution of the algorithm. The pseudo code of the improved GA-PSO algorithm is described in Algorithm 1, where a single point crossover operation with the rate of 0.9 is employed and the mutation rate is set to be 0.5.

4.3. Assembly Posture Optimization and Deviation Calculation. Based on the improved PSO algorithm, an optimization mathematical model under certain constraints is solved by Matlab®. Since PSO is a stochastic algorithm, statistical method was applied in the experiment to verify the reliability of the results, as adopted in literatures [29, 34]. Considering the computational cost, we performed 20 independent runs with the same algorithm parameters and the same input data. The statistical results, including the best, median, worst, mean, and standard deviation values of the objective function over the 20 runs, are summarized in Table 1. Besides, the results of the improved algorithm were also presented. The best feasible optimal solution of two algorithms (i.e., the final posture adjustment parameters) is presented in Table 2, which could guide the adjustment of the actual part. In order to analyze the results of the previous PSO algorithm and the improved algorithm, we also performed a Wilcoxon test on the results in Table 1. The test has been performed against the null hypothesis that there is no difference in their population means. Under this null hypothesis, the Wilcoxon rank sum test (Wilcoxon-Mann-Whitney test) [45] has been performed

- (1) Initialize the population and calculate the fitness so as to find the optimal position P_{best_i} of particle i and the global optimal position G_{best}
- (2) Set evolution times g_{max} and initialize the current iteration $g = 1$
- (3) **do**
- (4) Update the velocity vector of every particle according to Eq. (6)
- (5) Update particle position according to the position equation Eq. (7)
- (6) Embed crossover and mutation operations
- (7) Initial probabilities of crossover (p_{cross}) and mutation ($p_{mutation}$)
- (8) Generate new solution by crossover and mutation
- (9) if $p_{cross} > \text{rand}$, crossover; end if
- (10) if $p_{mutation} > \text{rand}$, mutate; end if
- (11) Accept the new solution if its fitness increases
- (12) Update the P_{best_i} of particle i and the global optimal position G_{best}
- (13) **While accuracy requirement or maximum iterations is not satisfied**

ALGORITHM 1: Pseudo code of the improved GA-PSO algorithm.

TABLE 1: Statistical comparisons for the experiment.

| Best | Median | Worst | Mean | Standard deviation |
|---------------|-----------|-----------|-----------|--------------------|
| (1) 84.991846 | 85.505687 | 87.378064 | 85.953390 | 1.060213 |
| (2) 83.033714 | 83.258915 | 84.182042 | 83.472345 | 0.532362 |

(1) previous algorithm; (2) improved algorithm

TABLE 2: The posture adjustment parameters.

| T_x/mm | T_y/mm | T_z/mm | α/rad | β/rad | γ/rad |
|---------------|----------|-----------|--------------|-------------|--------------|
| (1) -2.422025 | 7.042101 | -5.130164 | -0.003009 | 0.002831 | -0.003804 |
| (2) -3.879553 | 8.697785 | -7.096523 | 0.000597 | -0.001751 | -0.001828 |

(1) previous algorithm; (2) improved algorithm

TABLE 3

(a) Wilcoxon rank sum test: Ranks of two methods

| Method | N | Mean Rank | Sum of Ranks |
|-----------------|----|-----------|--------------|
| Improved Method | 20 | 10.50 | 210.00 |
| Previous Method | 20 | 30.50 | 610.00 |

(b) Wilcoxon rank sum test: Test statistics of two methods

| Mann-Whitney U | Wilcoxon W | Z | Asymp. Sig. (2-tailed) | Exact Sig. [2*(1-tailed Sig.)] |
|----------------|------------|--------|------------------------|--------------------------------|
| .000 | 210.000 | -5.410 | 6.3018E-8 | 1.4509E-11 |

in Table 3. It can be seen that the z value is -5.410, and p value is less than 0.001. Thus, the means of these two methods differ significantly. Since the mean of the improved algorithm is better than the previous PSO algorithm, the improved one has better results and global search ability.

After that, the virtual assembly process of the whole structure is simulated by CATIA®, as shown in Figure 10(a). The simulation models, which include the wing box body and the assembly tooling, were designed. Based on the posture transformation parameters which were output from Matlab®, the posture of the panel can be adjusted through the original function of CATIA®. In this experiment, the adjustment of the panel can be realized through three

six-degree-of-freedom locators. Once the optimal posture of the panel is obtained, these three locators will adjust their position to fit the panel accurately. The posture transformation parameters can be transmitted from the virtual assembly to the real assembly with the help of Optical Tooling Points (OTPs). OTPs are defined during the tooling design, which is used to control the spatial position of assembly tooling locators and they can record the movement of the locators. In the real assembly, a laser tracker is used to track the position of OTPs. After these OTPs are adjusted to the target position, the panel only needs to be positioned to fit the locators; then the current posture of the panel is already the target posture. After the panel was located on the target position,

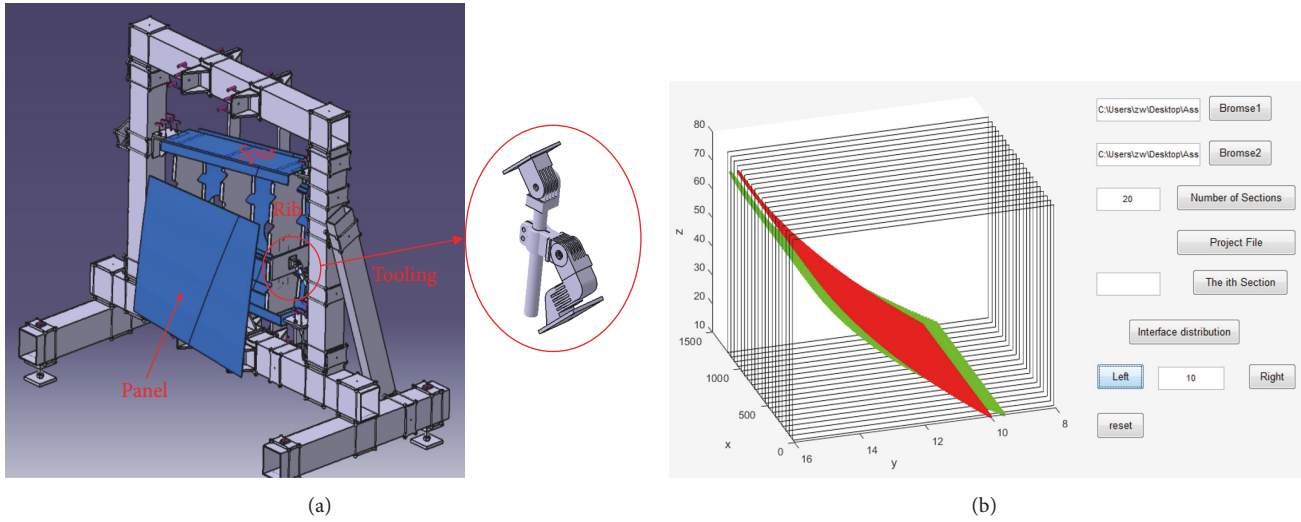


FIGURE 10: Assembly simulation: (a) virtual assembly of the wing box and (b) assembly deviation calculation.

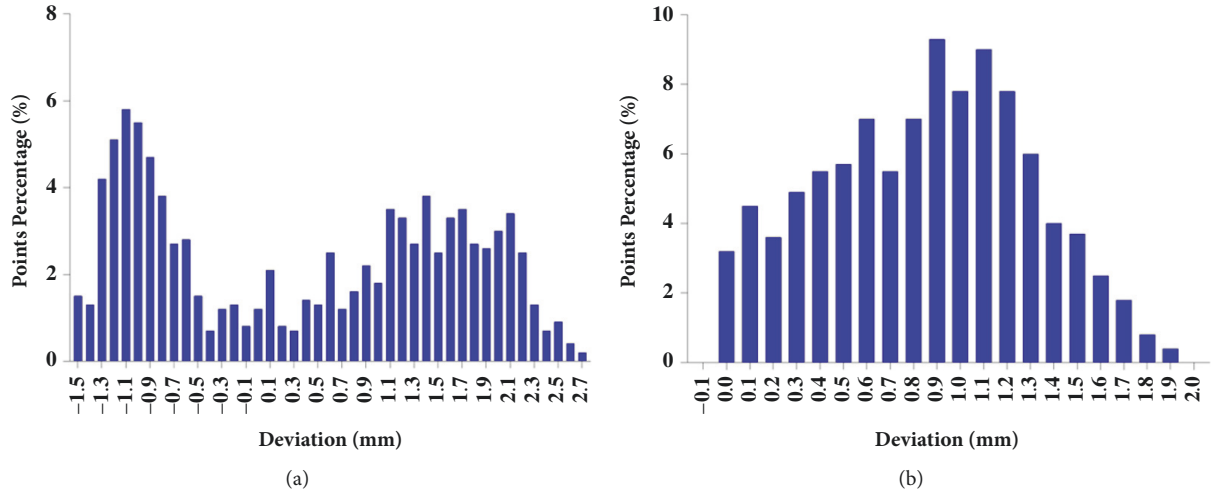


FIGURE 11: The percentage of deviation points: (a) nominal position and (b) optimal position.

the deviation between the mating surfaces was calculated on the basis of point cloud slicing method, as can be seen from Figure 10(b).

4.4. Comparison and Discussion. The assembly deviations of the panel and the skeleton before and after adjustment are calculated, respectively, by the improved algorithm, and the percentage of deviation points is counted, as illustrated in Figure 11. It can be seen that noncompliant assembly occurred when the panel was located on the nominal position; then the overall gaps and the amplitude of deviation were reduced after posture optimization. Moreover, the final aerodynamic shape of the actual panel after adjustment was also measured using a laser tracker, as can be seen from Figure 12. The final shape deviation fluctuates slightly but is within the range of tolerance, which means that, by using the weighted distance objective function and constrained contact states, the final posture tends to be acceptable.

5. Summary and Outlook

Geometric deviations inevitably exist in every manufactured part and even a tiny form error may induce noncompliant assembly. However, only the orientation and position deviations are considered in most models in analysis of geometric variation rather than the form errors. This paper developed an approach for the assembly simulation and posture adjustment based on the skin model, which enables the prediction of the assembly deviation caused by part geometric errors.

In this method, the mechanical assembly of rigid parts with manufacture deviation is investigated and a numerical optimization method for the posture adjustment is employed. For the positioning of the skin model, an assembly feature based registration method is applied to improve the accuracy and efficiency of point cloud registration. Then the PSO algorithm is employed to solve an optimization mathematical model so as to search the optimal posture of

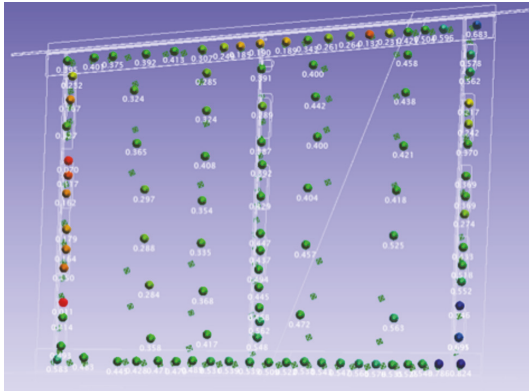


FIGURE 12: Measurement points on the actual part.

the assembled part. The numerical method is validated with experimental data obtained from a real wing box assembly in a production floor. Afterward, the assembly deviation on the mating surfaces is calculated based on the point cloud slicing method, which could quantitatively evaluate the contact quality of the multiple parts. Before computing the deviation, the mating surfaces of the assembled parts are extracted according to the change of the normal vector at the boundary regions. Finally, a case study of the aircraft wing box scaling model is conducted. The panel is adjusted and located according to the final transformation matrices, and the assembly deviations of the skeleton and the panel are calculated. It can be seen that, after adjusting, the overall gaps between the assembled parts become uniform, which could facilitate later assembly process, and the final shape of the structure is within the tolerance. This paper discussed the assembly simulation and part posture adjustment on the basis of point clouds, which is effective in reducing the handling work (such as fettling or shimming) to maintain the interface tolerances. However, this simulation model simplified the posture adjustment as rigid transformation; more complex assembly processes considering the bolt fastening force and part deformation need to be combined with the skin model representation in future research works. The authors are already working on the extension of the methodology employed in this paper to more complex nonrigid assembly situations. In future work, the issues of more different assembly deviation calculation methods will also be addressed.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work is supported by the National Science and Technology Major Project of the Ministry of Science and Technology of China (Grant No. 2014ZX04001071).

References

- [1] Y. S. Hong and T.-C. Chang, "A comprehensive review of tolerancing research," *International Journal of Production Research*, vol. 40, no. 11, pp. 2425–2459, 2002.
- [2] L. Picheral, K. Hadj-Hamou, and J. Bignon, "Robust optimization based on the Propagation of Variance method for analytic design models," *International Journal of Production Research*, vol. 52, no. 24, pp. 7324–7338, 2014.
- [3] U. Prisco and G. Giorleo, "Overview of current CAT systems," *Integrated Computer-Aided Engineering*, vol. 9, no. 4, pp. 373–387, 2002.
- [4] T. Zhang, Z. Zhang, X. Jin, X. Ye, and Z. Zhang, "An innovative method of modeling plane geometric form errors for precision assembly," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 230, no. 6, pp. 1087–1096, 2016.
- [5] E. Goka, L. Homri, P. Beaurepaire, and J. Dantan, "Geometrical variation simulation for assembly with form defects," in *Proceedings of the ASME 2017 International Mechanical Engineering Congress and Exposition*, vol. 2, Tampa, Florida, USA, 2018.
- [6] W. Polini and A. Corrado, "Geometric tolerance analysis through Jacobian model for rigid assemblies with translational deviations," *Assembly Automation*, vol. 36, no. 1, pp. 72–79, 2016.
- [7] J.-Y. Dantan, A. Ballu, and L. Mathieu, "Geometrical product specifications - model for product life cycle," *Computer-Aided Design*, vol. 40, no. 4, pp. 493–501, 2008.
- [8] I. 17450-1:2011, "ISO 17450-1:2011 - Geometrical product specifications (GPS) – General concepts – Part 1: Model for geometrical specification and verification," Iso.org, 2018.
- [9] M. Zhang, N. Anwer, A. Stockinger, L. Mathieu, and S. Wartzack, "Discrete shape modeling for skin model representation," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 227, no. 5, pp. 672–680, 2013.
- [10] N. Anwer, A. Ballu, and L. Mathieu, "The skin model, a comprehensive geometric model for engineering design," *CIRP Annals - Manufacturing Technology*, vol. 62, no. 1, pp. 143–146, 2013.
- [11] L. Homri, E. Goka, G. Levasseur, and J.-Y. Dantan, "Tolerance analysis — Form defects modeling and simulation by modal decomposition and optimization," *Computer-Aided Design*, vol. 91, pp. 46–59, 2017.
- [12] B. Schleich, N. Anwer, L. Mathieu, and S. Wartzack, "Skin Model Shapes: A new paradigm shift for geometric variations modelling in mechanical engineering," *Computer-Aided Design*, vol. 50, pp. 1–15, 2014.
- [13] B. Schleich and S. Wartzack, "Approaches for the assembly simulation of skin model shapes," *Computer-Aided Design*, vol. 65, pp. 18–33, 2015.
- [14] B. Schleich, N. Anwer, L. Mathieu, and S. Wartzack, "Contact and Mobility Simulation for Mechanical Assemblies Based on Skin Model Shapes," *Journal of Computing and Information Science in Engineering*, vol. 15, no. 2, 2015.

- [15] B. Schleich and S. Wartzack, "Novel approaches for the assembly simulation of rigid Skin Model Shapes in tolerance analysis," *Computer-Aided Design*, vol. 101, pp. 1–11, 2018.
- [16] B. Zhang, B.-G. Yao, and Y.-L. Ke, "A novel posture alignment system for aircraft wing assembly," *Journal of Zhejiang University SCIENCE A*, vol. 10, no. 11, pp. 1624–1630, 2009.
- [17] Y. Li, L. Zhang, and Y. Wang, "An optimal method of posture adjustment in aircraft fuselage joining assembly with engineering constraints," *Chinese Journal of Aeronautics*, vol. 30, no. 6, pp. 2016–2023, 2017.
- [18] H. Yukan, L. Yuan, Z. Jie, W.-B. Tang, and J. Shoushan, "A simple mechanical measurement system for the posture evaluation of wing components using the PSO and ICP algorithms," *Assembly Automation*, vol. 35, no. 1, pp. 104–113, 2015.
- [19] W. B. Tang, Y. Li, J. F. Yu, J. Zhang, and Y. L. Sun, "A posture evaluation method for aircraft wing based on a simple measurement system," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 227, no. 4, pp. 627–630, 2013.
- [20] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.
- [21] W. Van Holland and W. F. Bronsvort, "Assembly features in modeling and planning," *Robotics and Computer-Integrated Manufacturing*, vol. 16, no. 4, pp. 277–294, 2000.
- [22] S. Kaneko, T. Kondo, A. Miyamoto, and S. Igarashi, "Robust ICP registration algorithm extended by M-estimation," *Seimitsu Kogaku Kaishi/Journal of the Japan Society for Precision Engineering*, vol. 67, no. 8, pp. 1276–1280, 2001.
- [23] J. X. Dhôte, A. J. Comer, W. F. Stanley, and T. M. Young, "Study of the effect of liquid shim on single-lap joint using 3D Digital Image Correlation," *Composite Structures*, vol. 96, pp. 216–225, 2013.
- [24] S. Flöry and M. Hofer, "Surface fitting and registration of point clouds using approximations of the unsigned distance function," *Computer Aided Geometric Design*, vol. 27, no. 1, pp. 60–77, 2010.
- [25] C. A. Coello Coello, "Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art," *Computer Methods Applied Mechanics and Engineering*, vol. 191, no. 11–12, pp. 1245–1287, 2002.
- [26] R. C. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proceedings of the 6th International Symposium on Micromachine and Human Science*, pp. 39–43, Nagoya, Japan, October 1995.
- [27] Y. Shi and R. C. Eberhart, "Parameter selection in particle swarm optimization," *Lecture Notes in Computer Science*, vol. 1447, pp. 591–600, 1998.
- [28] R. C. Eberhart and Y. Shi, "Particle swarm optimization: developments, applications and resources," in *Proceedings of the 2001 Congress on Evolutionary Computation*, pp. 81–86, IEEE Press, New York, NY, USA, May 2001.
- [29] T. H. Kim, I. Maruta, and T. Sugie, "A simple and efficient constrained particle swarm optimization and its application to engineering design problems," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 224, no. 2, pp. 389–400, 2010.
- [30] M. Dorigo and L. M. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 53–66, 1997.
- [31] M. M. Eusuff and K. E. Lansey, "Optimization of water distribution network design using the shuffled frog leaping algorithm," *Journal of Water Resources Planning and Management*, vol. 129, no. 3, pp. 210–225, 2003.
- [32] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," *Journal of Global Optimization*, vol. 39, no. 3, pp. 459–471, 2007.
- [33] H. Ma, S. Ye, D. Simon, and M. Fei, "Conceptual and numerical comparisons of swarm intelligence optimization algorithms," *Soft Computing*, vol. 21, no. 11, pp. 3081–3100, 2017.
- [34] L. C. Cagnina, S. C. Esquivel, and C. A. Coello, "Solving engineering optimization problems with the simple constrained particle swarm optimizer," *Informatica*, vol. 32, no. 3, pp. 319–326, 2008.
- [35] R. S. Parpinelli, F. R. Teodoro, and H. S. Lopes, "A comparison of swarm intelligence algorithms for structural engineering optimization," *International Journal for Numerical Methods in Engineering*, vol. 91, no. 6, pp. 666–684, 2012.
- [36] K. E. Parsopoulos and M. N. Vrahatis, "Particle swarm optimization method for constrained optimization problems," in *Intelligent Technologies-Theory and Application: New Trends in Intelligent Technologies*, vol. 76, pp. 214–220, 2002.
- [37] S. Kitayama, M. Arakawa, and K. Yamazaki, "Penalty function approach for the mixed discrete nonlinear problems by particle swarm optimization," *Structural and Multidisciplinary Optimization*, vol. 32, no. 3, pp. 191–202, 2006.
- [38] R. E. Perez and K. Behdinan, "Particle swarm approach for structural design optimization," *Computers & Structures*, vol. 85, no. 19–20, pp. 1579–1588, 2007.
- [39] K. Sedlaczek and P. Eberhard, "Using augmented Lagrangian particle swarm optimization for constrained problems in engineering," *Structural and Multidisciplinary Optimization*, vol. 32, no. 4, pp. 277–286, 2006.
- [40] A. Homaifar, S. H. Y. Lai, and C. X. Qi, "Constrained optimization via genetic algorithms," *Simulation*, vol. 62, no. 4, pp. 242–254, 1994.
- [41] O. Kulkarni, N. Kulkarni, A. J. Kulkarni, and G. Kakandikar, "Constrained cohort intelligence using static and dynamic penalty function approach for mechanical components design," *International Journal of Parallel, Emergent and Distributed Systems*, pp. 1–19, 2016.
- [42] P. G. Maropoulos, J. E. Muelaner, M. D. Summers, and O. C. Martin, "A new paradigm in large-scale assembly-research priorities in measurement assisted assembly," *The International Journal of Advanced Manufacturing Technology*, vol. 70, no. 1–4, pp. 621–633, 2014.
- [43] C. Lartigue, A. Contri, and P. Bourdet, "Digitised point quality in relation with point exploitation," *Measurement*, vol. 32, no. 3, pp. 193–203, 2002.
- [44] W. G. Glen, W. J. Dunn III, and D. R. Scott, "Principal components analysis and partial least squares regression," *Tetrahedron Computer Methodology*, vol. 2, no. 6, pp. 349–376, 1989.
- [45] L. Baringhaus and D. Gaigall, "Efficiency comparison of the Wilcoxon tests in paired and independent survey samples," *Metrika*, vol. 81, no. 8, pp. 891–930, 2018.



2018-10-18

Posture optimization algorithm for large structure assemblies based on skin model

Zhang, Wei

Hindawi

Zhang W, An L, Sherar P, Tian W. (2018) Posture optimization algorithm for large structure assemblies based on skin model. Mathematical Problems in Engineering, Volume 2018, Article number 9680639

<https://doi.org/10.1155/2018/9680639>

Downloaded from Cranfield Library Services E-Repository